



生醫實驗-期末 FINAL

運動與熱量監控裝置

第六組

電機三 B00901107 林芷儀

電機三 B00901173 林子涵

電機三 B00901176 黃郁蓁

目錄

壹、產品計劃書	p2
貳、實作方法	
1. 實作_訊號前置處理 ECG 電路解析.....	p3~p4
2. 實作_使用 labview.....	P5~p7
3. 實作_熱量攝取紀錄.....	p9~12
參、操作步驟.....	p13
肆、結果分析與改善.....	p14
伍、願景.....	p15

壹、產品計劃書

A. 題目：自主熱量紀錄系統

B. 功能：紀錄生活中的運動，以及攝取的食物，提供個人化的熱量檔案。

C. 動機：

現代人生活普遍忙碌，很少人可以完全空出一段完整的時間去做運動，如果有一個配件可以記錄我們生活中的運動情形，哪怕只是為了趕時間而快走 15 分鐘的有氧運動，或是只是幫忙提了許多重物的肌力訓練，我們都可以將它紀錄下來，並在一天過後作分析，提供使用者對自己生活中的運動情形。搭配自主的熱量攝取紀錄，更可以成為維持良好體態與健康的整套配件。

D. 應用性：

希望能發展成方便配戴的無線產品，在不需要特別自主紀錄的情況下將身體的運動情形記錄下。

E. 市場性：

在騎腳踏車通勤被大力推廣之下，台灣民眾越來越注重運動養生與身體健康。然而年輕一代普遍生活忙碌，希望做的事情是有效率並有結果的。這項產品能夠在無意中針對消耗的熱量提出實際數據，搭配自主的攝食熱量紀錄，應該對許多注意身材的年輕女性具有不小的吸引力。

對忙碌又希望能維持健康的上班族也有一定的實用性。

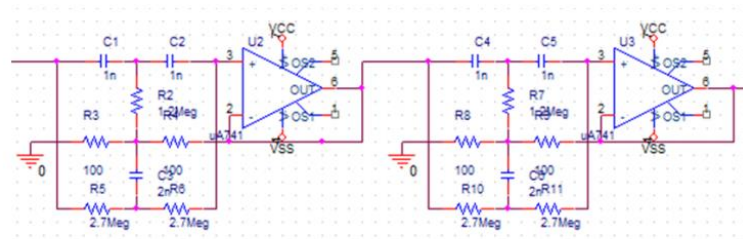
F. 使用之儀器、材料：

1. Labview
2. 測量心電與肌電訊號所需的配備
3. 軟體應用：C++、java

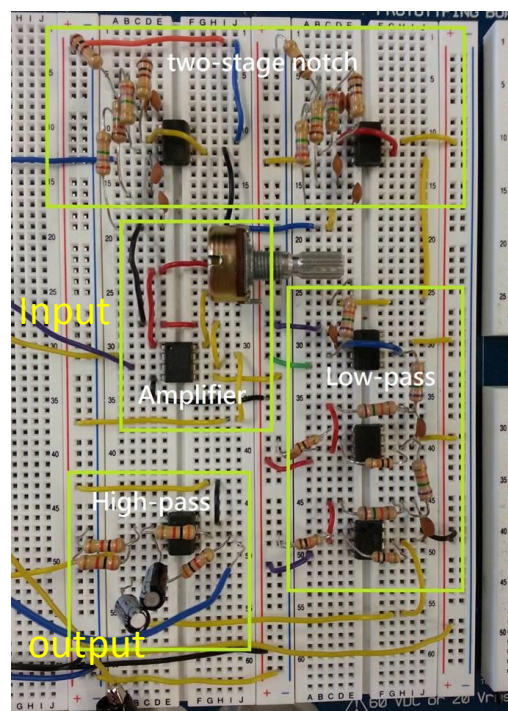
貳、實作方法

1. 訊號前置處理

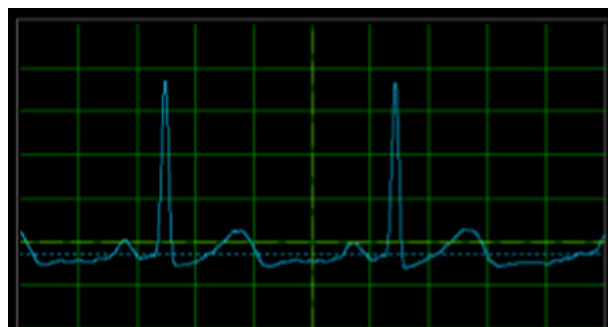
- 目的：有效放大微弱的心電訊號，並將非心電頻率的訊號濾掉，其中最主要的就是 60Hz 的電器雜訊。
- 方法：利用以下放大器以及濾波器串接來處理訊號。
 - 儀表放大器：使用 INA128， $\text{gain} = 400 \sim 500 \text{V/V}$
 - 帶拒濾波器(Notch)：將 60Hz 的雜訊消除，這裡我們使用了兩級，每一級在 60Hz 的地方都會壓抑 34dB，所以兩極總共是 -68dB。



- 以下為實際電路圖：

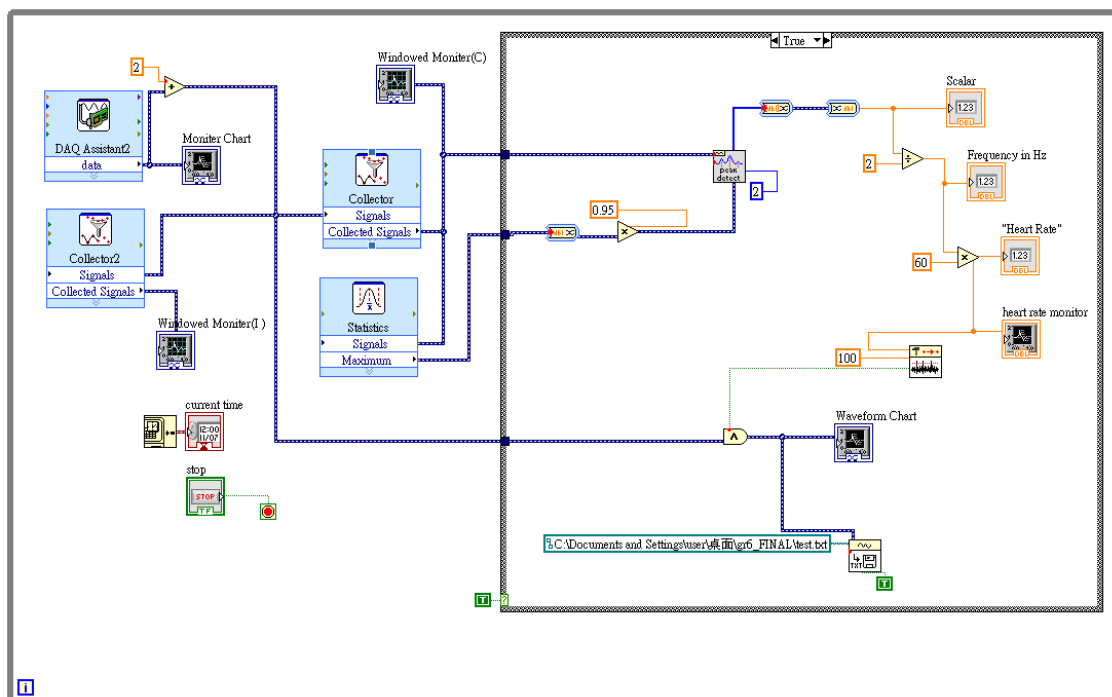


- 輸出結果：

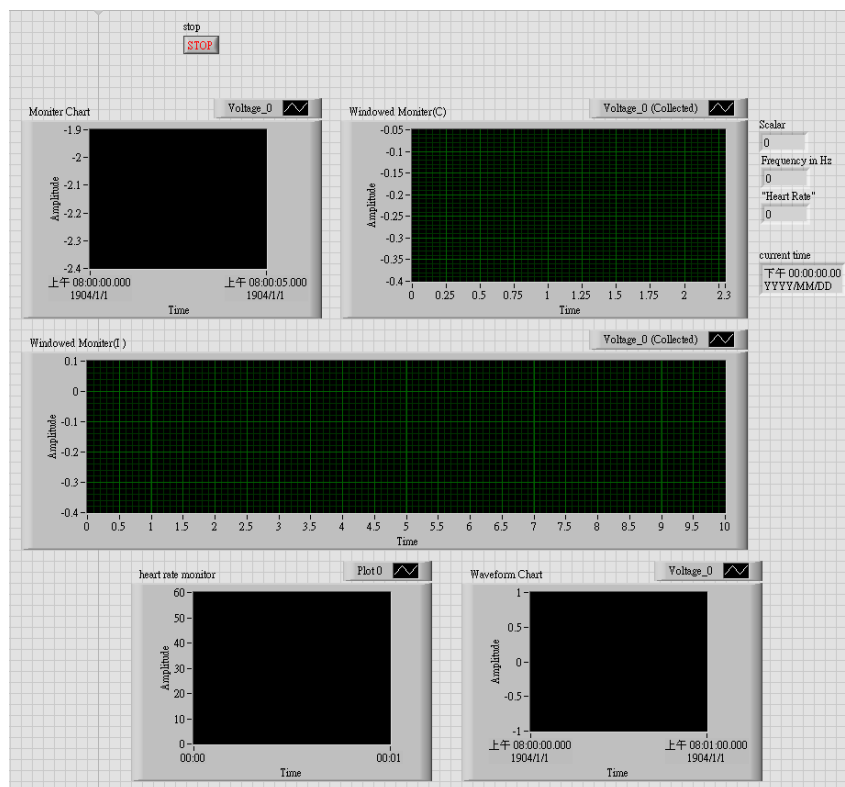


2. 使用 labview 算每分鐘心跳與偵測運動時間區間
 - a. 目的：算出每分鐘心跳數目，並記錄下 data 輸出為檔案。
 - b. 方法：

● Labview Block Diagram



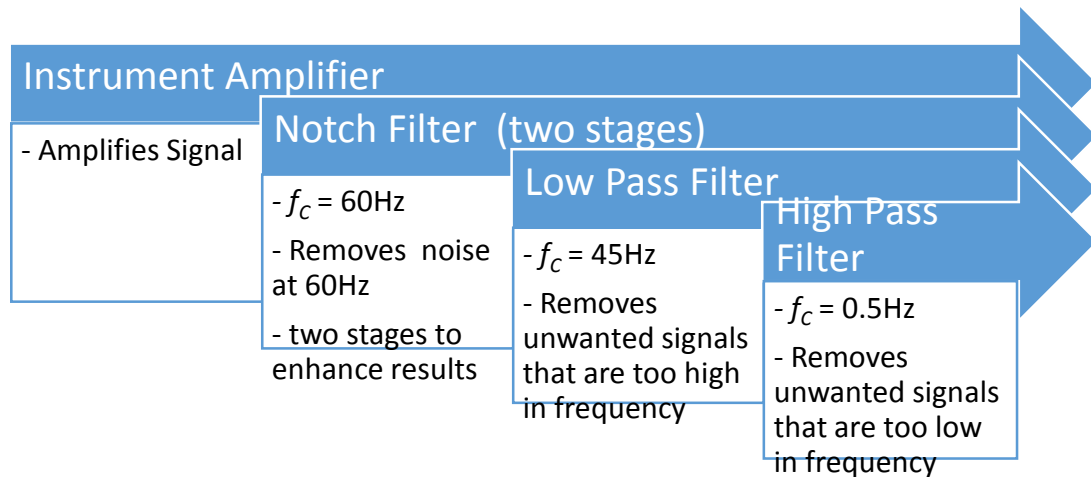
● Labview Front Panel



- DAQ Assistant

The DAQ Assistant acquires data from a circuit connected on a NI ELVIS II. The circuit is the same circuit we designed for Experiment 2 this semester. Here is an brief overview of the circuit design. For more information, please refer to the lab report for Experiment 2.

- Overview



- Labview Functions (Collector & Statistics)

- Labview Collector Function

Collects a signal at each time step of the simulation and returns a history of the signal value and the time at which this function recorded each value in the history.

- Collector

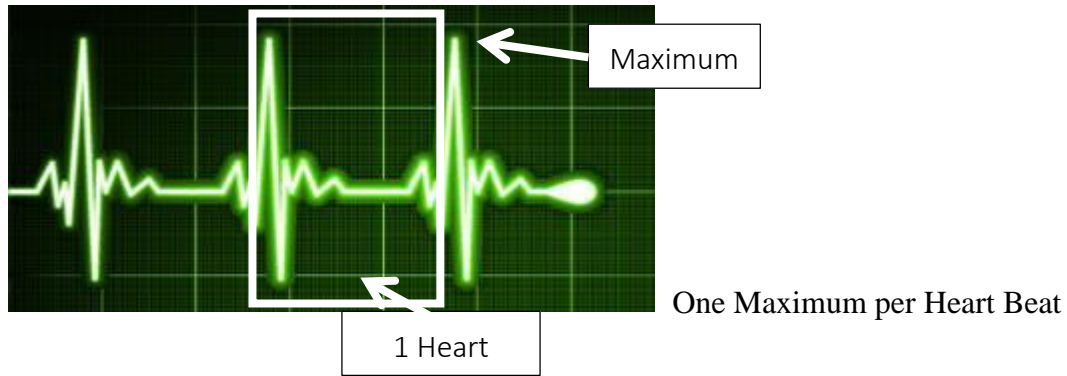
Collects the data for 2 seconds and output the data in “Windowed Monitor (C)” in the front panel.

- Collector2

Collects the data for 10 seconds and output the data in “Windowed Monitor (I)” in the front panel.

- Labview Statistics Function

Returns the selected parameter of the first signal in a waveform. Here, we used the statistics function to find the maximum (the highest point in a set of values in Signals) of our collected data from the Collector function (2 second of data).



● Obtaining Heart Rate

The Statistics Function determines the maximum point in a set of values. In each heart beat, the maximum value hypothetically appears once.

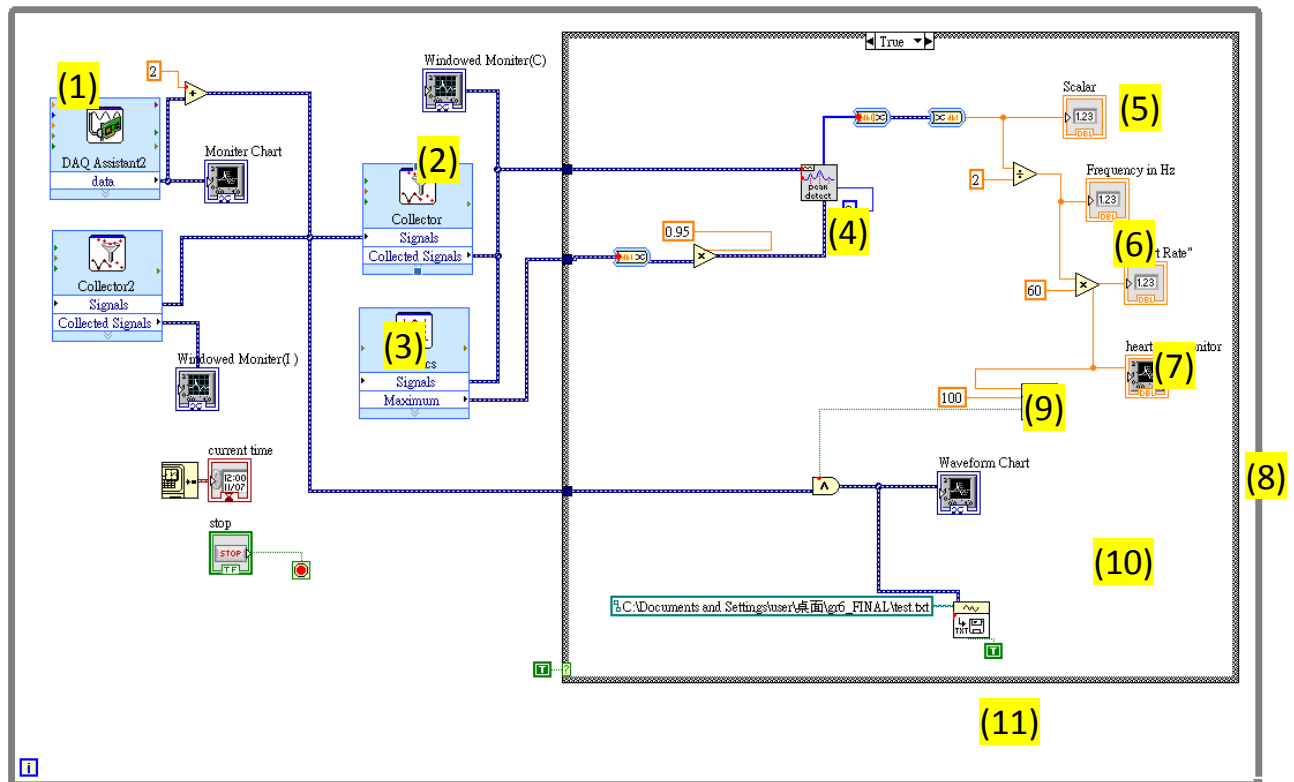
Using Peak Detect, the number of maximum points is identified. Any signal below 95% of the maximum value determined by the Statistics function is disregarded.

Frequency of Heart Beat (Hz) = number of peaks in the time period / number of seconds in the time period

Heart Beat Per minute = Frequency of Heart Beat X 60 seconds

In our example, the time period is set at 2 seconds. Thus, Frequency is the number of peaks divided by 2. Heart Beat is the Frequency multiplied by 60.

● Summary



- (1) DAQ Assistant2 obtains data from the circuit on NI ELVIS II.
- (2) The signal is collected by the Collector function for a period of 2 seconds.
- (3) Using the Statistics function, the maximum value in that time period is determined.
- (4) Using Peak Detect, the number of peaks in a time period is identified. Any signal below 95% of the maximum value determined by the Statistics function (in step 3) is disregarded.
- (5) The number of peaks is displayed as a SCALAR.
- (6) Since the time period is 2 seconds, the scalar (in step 5) is divided by 2 to obtain the frequency of the heart beat in Hertz.
- (7) There are 60 seconds in a minute, so the frequency (in step 6) is multiplied by 60 to obtain the heart rate in beats per minute.
- (8) The heart rate (in step 7) is graphed onto the “heart rate monitor” to observe the changes in heart rate over time.
- (9) We set the borderline of the heart beat when exercising to be 100. In other words, when the heart rate is over 100 beats per minute, the output would be “1”. Otherwise, the output is “0”.
- (10) The output of “0” and “1” is graphed onto the “Waveform Chart” to observe when the subject is exercising throughout the day.
- (11) The output is recorded overtime and saved in the form of a txt file.

3. 熱量攝取紀錄

a. 目的：讓使用者可以紀錄一天所攝食的熱量。

b. 方法：

- 主平台：主平台是一個 class “Database”。

主食類	飲料類	水果類
白麵包	金車麥根沙士	荔枝
Name: 白麵包	Name: 金車麥根沙士	Name: 荔枝
Unit: 1片(25g)	Unit: 350ml	Unit: 4粒
Calorie: 75	Calorie: 133	Calorie: 45

1. 輸入

a. 個人資料

- 使用者將"姓名"、"身高"、"體重" 輸入對應欄位，並按下存入檔案。系統 會用之初始化 Database。其中會用到 "體重" 作為熱量消耗的運算標準。

b. 進食

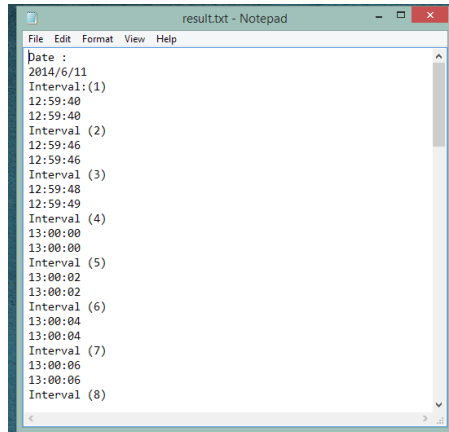
- 使用者進食後，將"食物名稱"、"單位"、"熱量" 輸入對應欄位，並按下存入檔案。系統會將之視為一次的資料包裝成一個 "FoodProfile" Object，將之推進 Database 的"FoodList" Vector 中。

2. 食物 vs 熱量查詢系統

- 這是一個 class “SearchSystem”。
- 共包含三個選單，每個選單代表一個 Vector；選單內的選項代表 Vector 內的物件，物件為 "FoodProfile" Object。
- 我們將蒐集到的食物和熱量的資料整理成 “FoodProfile” Object，分類後塞進不同的 Vector，建置成資料庫。往後若要再增加分類或新增食物，只要再新增 Vector 或 "FoodProfile" 即可。亦考慮將 "SearchSystem” 做成開放式的，可以讓使用者自由製作。

- 使用者選擇其中一個選項 (食物) 後，食物的"名稱"、"單位"、"熱量"會顯示在選單下方。同時也會有貼心的小動作，自動寫入使用者輸入處的欄位上。但使用者還保有更改的空間，因為按下"存入檔案" 資料才會確實寫入紀錄。

3. Run Me

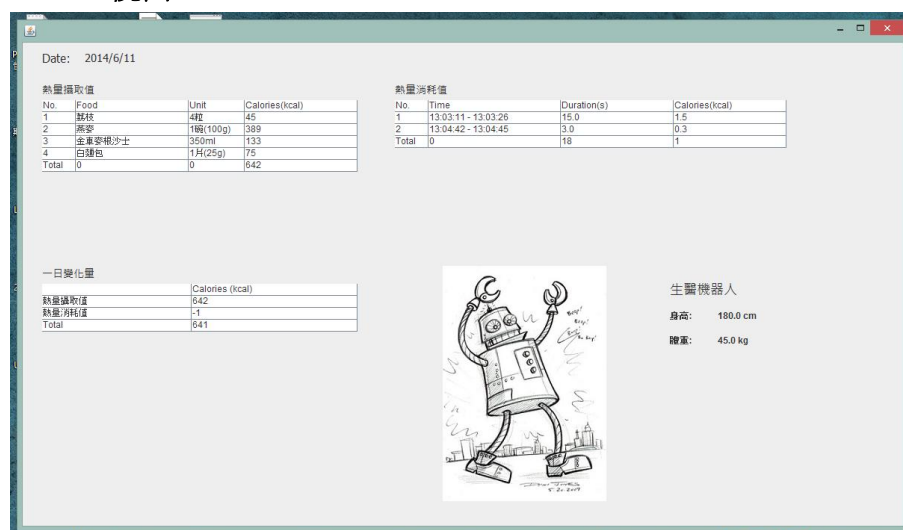


- 此按鈕會運行 "BioMed.exe"，並整理出 "result.txt"
- 跑完 Labview 會產生一個 "test1.txt" (詳見 Labview)。“BioMed.exe” 會讀取此文字檔，a 每一秒看他有多少個 '1'。如果 '1' 比 '0' 多就把那一秒的值設為 '1'。然後擷取 '1' 時段內的初始時間和結束時間，寫成一個一個 Interval，再輸出成 "result.txt"。文件的開頭會寫入當天的日期。

4. Print Record

- 此按鈕會製作出 “Record” 視窗
- 同時輸出 “Record.txt”

Record 視窗



1. 日期

- 由"result.txt"的第二行讀取。

2. 熱量攝取值

- 由“FoodList” Vector 讀取，將對應資料輸入個別欄位，製作成表格。
- 最後一列會計算出 Calories 的 Total。

3. 熱量消耗值

- 由“result.txt”讀取一時段，算出時段內的時間長度，若小於“threshold”(運動最低時間長度，demo 時設定為 3 秒) 則放棄資料。若時間長度通過“threshold”，則套入“運動 vs. 熱量轉換公式”(demo 時設定運動為慢跑，體重為 50kg，每秒消耗 0.1kcal)，計算出熱量消耗。同樣將對應資料輸入個別欄位，製作成表格。
- 最後一列會計算出 Duration 和 Calories 的 Total。
- “運動 vs. 熱量轉換公式”：時間*體重*消耗熱量 (大卡/每公斤體重/每(時間單位))

運動項目	消耗熱量	運動項目	消耗熱量	運動項目	消耗熱量
搖呼拉圈	2.3	游泳(0.4公里/小時)	4.4	滑雪(15公里/小時)	7.2
拉筋運動	2.5	有氧舞蹈	5.0	溜直排輪	8.0
騎腳踏車(8.8公里/小時)	3.0	羽毛球	5.1	手球	8.8
下樓梯	3.1	排球	5.1	跳繩(60-80下/每分鐘)	9.0
走路(4公里/小時)	3.1	跳舞(快)	5.1	慢跑(145公尺/每分鐘)	9.4
高爾夫球	3.7	溜輪鞋	5.1	騎腳踏車(20.9公里/小時)	9.7
保齡球	4.0	乒乓球	5.3	拳擊	11.4
快步走(6.0公里/小時)	4.4	溜冰刀(16公里/小時)	5.9	划船比賽	12.4
上樓梯	4.4	網球	6.2	跑步(16公里/小時)	13.2
划船(4公里/小時)	4.4	爬岩(35公尺/小時)	7.0		

消耗熱量：大卡/每公斤體重/每小時

活動項目	消耗熱量	活動項目	消耗熱量	活動項目	消耗熱量
泡澡	84	洗衣服	57	坐公車(站著)	33
打掃	114	講電話	33	坐公車(坐著)	53
插花	57	睡午覺	24	燙衣服	60
郊遊	120	洗碗	68	念書	44
散步	66	逛街	55	買東西	90
看電視	36	開車	41	看電影	33
唱KTV	40.5	遛狗	65		

消耗熱量：大卡/每三十分鐘

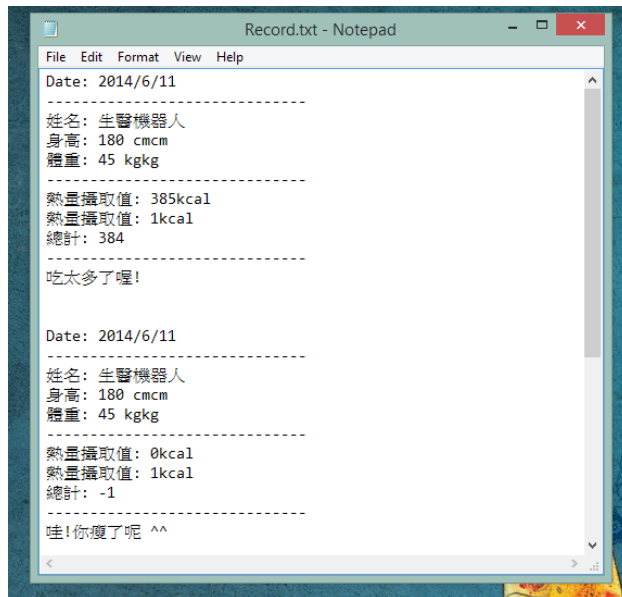
- 依據上述資料，“threshold”應會設定為 30 分鐘。
- 未來希望能找出心跳速率和熱量

消耗率的直接關係，作為“運動 vs. 熱量消耗轉換公式”。

4. 一日變化量

- 取熱量攝取值和熱量消耗值相減得出一日的熱量淨變化量。

Record.txt



- 包含“日期”、“個人資料”、當日的“熱量攝取值”、“熱量消耗值”、“總計”。
- 為“更新式”紀錄法。對於同一個使用者，新的資料會寫在舊資料的後面，但在同一個文件裡，做成一個歷史紀錄。

參、操作步驟：

- (1) 將貼片貼上 ECG 測量時的位置，將線接好，等待約 2~3 分鐘等訊號穩定。
- (2) 將 labview 打開，turn on。
- (3) 將 labview 最後丟出的.txt 放到 java 紀錄熱量攝取的程式的資料夾中。
- (4) 輸入個人資料、飲食攝取、分量。
- (5) 按下 botton 計算一整天下來的總熱量攝取與熱量消耗。

肆、結果分析與改善

● Labview 部分

最後我們在 demo 的時候，發生了最不想發生的狀況，就是 **baseline** 的漂移，因為 **baseline** 飄移所以在 **labview** 那邊計算心跳時有稍作調整。

因為 **base line** 漂移的問題如下。也就是為了避免因為 **base line** 往上漂移而使那一段時間的 **peak** 過大所以 **threshold** 也過大，最後在 **peak detection** 的時候，除了一個 **peak** 外，其他的訊號都不算，造成心跳變得異常慢。反之，若 **base line** 往下漂移，則是會讓那段區間的 **peak** 值過小，使其他訊號都算是超過 **threshold**，造成心跳異常的大。

我們當初解決方式是取較小的時間區間，因為 **base line** 在短時間內不會有太大的改變(頻率較小)，所以當初決定用這個方式解決，且自己在測試的時候效果也不錯。但這次漂移的實在太嚴重，所以還是發生了很大的誤差。

助教跟我們說，最有效的方式並不是在處理訊號分析的時候做調整，而是在抓到訊號的時候就要夠準確。於是我們觀察 **baseline** 的漂移訊號，發現這個雜訊的頻率是 **0.2Hz**，然後如果假設心跳的下限為每分鐘 **50** 下，那麼其對應的頻率就是 **0.8Hz**，所以可能有效的改善方式是將高通濾波器的 **f0** 改到大約 **0.8Hz** 的地方以加強雜訊的壓抑。把 **base line** 漂移的問題解決掉之後，就可以直接以 **60** 秒為單位去算心跳，而不會因為取太小的時間區間而造成不準確。

● 熱量攝取記錄部分

這部分其實還是有很大的發展空間，譬如像讓使用者與機器的介面更人性化與方便一點。參考方向有以下二：

- (1) 如果想要讓 **data base** 更有效一點，我們可以讓使用者個人化自己的食物 **data base**，因為根據調查，一個人所吃的食物其實沒有如此廣泛，是有侷限在某一範圍裏面的，透過這種方式我們可以省去一些不必要的資料儲存空間。
- (2) 如果可以讓使用者只須要拍張照，我們就可以判定這份食物是什麼，份量是多少，然後在 **data base** 裡面找他的相關資訊來計算，那將會是一個非常棒的使用介面，如此我們必須涉略更多關於影響辨識的知識，整個系統會變得更複雜卻更強大。

伍、願景

這個裝置其實應該要作為隨身攜帶的裝置，原本產品提案的時候就是希望「能在不知不覺中，完整的紀錄一個人一天的運動狀況與飲食狀況。」所以在實作的過程中，有兩大訴求就是「方便性」與「準確性」。

運動紀錄的部分其實有很多挑戰，我們做的熱量計算適用慢跑的熱量消耗去運算，如果能發展成適用於各種運動，那將會是很大的特色。但怎麼從心電訊號知道是哪種運動呢？我想這部分有些困難，可能要搭配一些肌電訊號加以輔助才能判別出現在的運動是哪種運動。另外，能加入肌電訊號也是目標，因為除了有氧運動，現在很多人也開始從事重量訓練。因為肌肉出力時所消耗的熱量也能夠十分可觀，所以除了能夠紀錄來自有氧運動的心電變化，若能加上肌力訓練的熱量消耗，這整個裝置將會變成一套完整的運動監視與紀錄工具。

現在市面上有很多飲食紀錄的 App，但自己的使用經驗往往是不夠方便，所以一開始可能會很勤勞的去紀錄，但時間一久就會完全荒廢。原因不外乎就是因為現代人的生活忙碌，沒有太多時間與耐心去一個個搜尋自己攝取的飲食與分量再去輸入，所以在飲食紀錄的討論那邊所提到的使用拍照的方式，是非常吸引客群的，當然，準確度的部分可能需要很強大的比對運算才能有一定水準的表現。

兩大部分都還有很大的進步與發展空間，若能真的發展為可用的市面產品，對於繁忙卻又逐漸講究身體健康的現代人而言，應該會是一個十分受歡迎的產品。